Automatic Section Segmentation of Medical Reports

Paul S. Cho, Ph.D.,[1] Ricky K. Taira, Ph.D., [2] and Hooshang Kangarloo, M.D. [2]

[1]Department of Radiation Oncology, University of Washington, Seattle, WA
[2]Medical Informatics Group, University of California, Los Angeles, CA

Abstract

Automated segmentation of medical reports can significantly enhance the productivity of the healthcare departments. While many algorithms have been developed for document summarization, passage retrieval, and story segmentation of news feeds, much less effort has been devoted to parsing of medical documents. We present an algorithm specifically developed for medical applications. The algorithm consists of two components. First, a rule-based algorithm is used to detect the sections that contain labels. It utilizes a knowledge base of commonly employed heading labels and linguistic cues seen within training examples. The second part of the algorithm handles the detection of unlabeled sections. It uses a combination of lexical pattern recognition and a classifier based on an expectation model for a particular class of medical reports. The proposed method was evaluated on three test corpora containing a total of 129,303 report sections. The detection rates for labeled and unlabeled sections for individual corpus ranged from 97.4% to 99.4% and from 96.5% to 99.0%, respectively. The rule-based approach is particularly effective for medical reports due to inherently structured nature of these documents.

INTRODUCTION

Segmentation of medical reports into topically cohesive sections is an essential task in patient information gathering and dissemination. Medical document retrieval systems can improve their indexing by knowing which parts of the report are relevant for specific types of queries. Clinical workstations could provide more elegant means of visualizing long and/or numerous medical reports for a patient if section breaks are known. Often, specific users are interested only in a subset of the text fields within a report. For example, a clinician may wish only to see a diagnostic conclusion section of a pathology or radiology report. An administrator may be only interested in the study description and reason for request sections. A coding system that uses natural language processing would benefit greatly by knowing which sections contain subjective (e.g., “Chief Complaint”) versus objective (e.g., “Findings”) patient descriptions An automated section extractor would also be useful in rapid generation of medical reports. Static data such as personal identification, history of illness, familial information, etc. are usually repeated in serial reports. An intelligent reporting system would automatically create a template for an existing patient with the static data already in place. Such a system would allow physicians to dictate only the new information.

Medical reports generated today are rarely formatted such that structural boundaries are known to a computer program. One reason is that the requirement for manual tagging of section boundaries would reduce transcription throughput. It would also require that a consensus be developed to match the target sections implied by the dictating physician. The problem is the same for speech recognition systems, which would require the physician to dictate the specific section names and that these section types be known by the speech recognition system. These may be steps that could again slow throughput and disturb the concentration of the dictating physician.

A plethora of algorithms has been proposed for computerized text segmentation. Skorochod’ko examined the degree of word overlap among the sentences to determine lexical connectivity [1]. Likewise, Halliday and Hansan utilized vocabulary similarity measures [2]. Morris and Hirst advanced the theory of lexical coherence and developed a thesaurus-based method to form lexical chains from which texts were structured [3]. Kozima proposed a semantic network to compute lexical cohesiveness between words [4]. Reynar introduced a graphical technique called *dotplotting* that detected topic boundaries by observing word repetition [5]. Hearst developed the *TextTiling* algorithm which utilizes patterns of lexical co-occurrence and distribution to detect changes in subtopics [6,7]. Also use of cue words to detect section transitions has been explored by some investigators [8,9]. For updated bibliography of works in the past decade see Pevner and Hearst [10].

Text segmentation algorithms have been applied to passage retrieval [11], automated summarization [12], genre detection [13], and story segmentation of news feeds [14]. However, none of the previously published methods was developed specifically for medical documents. This paper describes an algorithm designed to automatically partition a free text medical report into its constituent sections.

METHODS

Presently there is no universal standard or format for written medical reports in the U.S. While there are similarities, each institution, department, and individual physician has a unique policy and style of reporting. After examining a large number of reports from multiple institutions, it was decided that a supervised learning approach with its ability to adapt to local features would be most suitable for the task at hand.

During dictation it is customary for the physician to preface each section of the report with an appropriate heading such as “history”, “procedure”, “findings”, etc. Subsequently, these cue words are detected by the transcriptionist. Report structure intended by the author is then encoded into written document by insertion of section labels. Most commonly the section labels are written in upper-case characters followed by a colon. The section headings, however, are occasionally omitted by the dictating physician or inadvertently missed by the transcriber. Another clue of section boundary is provided by the transcriptionist who may insert paragraph breaks between sections. However, some favor faster wrap around typing style without insertion of hard carriage return throughout the document. The structure of the report may also be apparent from the document category, which is often included in the header. For example, a report may be an inpatient note, a discharge summary, an operation report, a procedure note, an outpatient consultation, or a letter. Depending on the category there are expected sections such as “Interval Events”, “Hospital Course”, “Discharge Diagnosis”, “Anesthesia”, and “Requesting Physician”. Individual idiosyncrasy is another clue. Some physician may like to open certain section with a certain phrase.

Features that characterize section boundaries as described above are extracted from a set of training examples according to report type (defined at the level of department). Quality and quantity of training samples are of utmost importance. If examples are erroneous, this introduces noise in the data used for modeling. Obtaining sufficient training examples for the complete spectrum of patterns (feature space) seen for a particular report structure is also critical. If the training examples do not include many patterns seen in future examples, it may be likely that the classifier can become over-trained; it has been intensively optimized on a training set distribution that is likely to be different than future test samples. The specialization of our section boundary detector by report type thus greatly improves the stationary expectation assumption. It also substantially reduces the dimensionality and complexity (number of independent parameters) as compared to one monolithic section boundary detector.

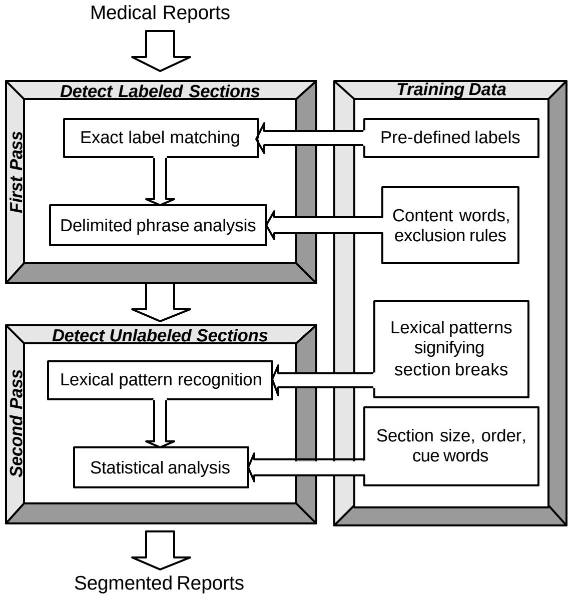
Medical Reports
Detect Labeled Sections
First Pass
Exact label matching
Delimited phrase analysis
Training Data
Pre-defined labels
Content words, exclusion rules
Detect Unlabeled Sections
Second Pass
Lexical pattern recognition
Statistical analysis
Lexical patterns signifying section breaks
Section size, order, cue words
Segmented Reports

Figure 1. Flow diagram of the section segmentation algorithm.

We use a graphical interface to generate training examples. The user selects a training report and manually specifies each section break and indicates their label. From the training data, we accumulate a list of all unique section labels. We then manually assign these labels to a class. For example, the labels “Conclusion” and “Impression” are both assigned to the class “CONCLUSION”. We then use this mapping to update all training examples with their assigned categorization. A list of *content* words are also compiled. These are characteristically abstract category words such as “assessment”, “comparison”, and “evaluation” that are frequently used in section labels.

The central problem we face is the construction of a model that can explain the observed facts indicated by the training examples. We divide the algorithm into two steps. First a rule-based algorithm is employed to detect the start of a new section. It uses a knowledge base of commonly employed heading labels and linguistic cues seen within training examples. However, it is not uncommon to have missing section-heading labels within a medical report. The second pass of the algorithm handles the detection of section boundaries in this case. It uses a combination of lexical pattern recognition and probabilistic classifier based on an expectation model for the document structure of a particular class of medical reports. The report class type is defined by the institution, the department, and the document type.

First Pass

The first pass attempts to find the beginning of all sections that have a heading label. The algorithm locates all occurrences within the report that match any of the labels found in the training set. These are considered as strong evidence for being candidate section headers. Other candidates are also included by searching for all occurrences of a colon.

Exact label matching

Character string matching of section heading candidates ending with a colon has high probability of yielding true positives. In matching a label candidate, care must be taken to consider possible variants that have not yet been identified in the training set. These variants often result from misspelling (e.g., “IMUNIZATION” instead of “IMMUNIZATION”) or case selection (e.g., “Immunization” vs. “IMMUNIZATION”). To overcome this problem, a normalization process is applied to the phrases being compared:

1. Remove special characters (e.g. apostrophe, hyphen, asterisk, parenthesis).
2. Convert to all upper case.
3. Remove vowels.
4. Remove consecutively repeated letters.

Delimited phrase analysis

Other phrases that end with a colon but do not match exactly with the learned samples are examined further. First, the candidate phrase located to the left of the delimiting colon is extracted. The candidate sentence is assumed to begin after one of the following delimiters: line break, period, colon, semicolon, or tab. Next, the label candidates are evaluated by a series of boolean operators:

1. Is the colon used to show time?
2. Is the colon used to show ratio?
3. Is the colon used to show a list?
4. Is the candidate phrase all capitalized?
5. Is the candidate phrase in title format?

The results are passed on to a classifier along with other parameters including the number of words and the number of *content* words in the phrase.

Second Pass

The task of report segmentation is complicated by the occasional absence of section labels. This could be caused by mistake or intentional omission. The second pass searches for unlabeled sections using two strategies: lexical pattern recognition and statistical evaluation.

Lexical pattern recognition

In the training examples one may observe lexical patterns that can be used reliably to locate hidden sections. Phrases that are used repeatedly can be detected by computing histograms of words and phrases across the corpus of training documents.

Example 1: In one corpus the phrase “Thank you for referring…” appeared at high frequency. Furthermore, the acknowledgement phrase always preceded the signature. In another corpus, the term “MD5 CHECKSUM” always followed the signature. Although the signature fields were never labeled by the transcriber these lexical patterns were used to segment the signature section.

Example 2: A certain physician always began the “PATIENT IDENTIFICATION” section with the same opening phrase, “This patient” or “The patient.” In the event that the IDENTIFICATION section is expected but not found in a report dictated by this particular physician, it is possible to detect IDENTIFICATION section by activating a pattern search in the vicinity of the report where the expected section is likely to be found.

Example 3: Prefix-suffix patterns were observed in reports from a department. Transcribers’ initials were invariably enclosed within one of the following character pairs: {{"-", "-"},{"(", ")"},{"¬", "|"},{"/", " "},{"/", "\n"},{"\n", "/"},{"\r", "/"},{"~", "~"}} These patterns were used to extract the initials information.

Statistical analysis

For a given class of reports there usually exist an essential set of section categories. If any of these sections remain undetected by the algorithms described thus far, an expectation model is used to locate the hidden sections.

Table 1. An example of section size statistics collected from 549 chest radiology reports. Units are in characters.

Section Label	Mean Length	Std Deviation
Report Header	378	74
Procedure	371	154
Procedure code	30	0.01
Comparison	21	7
History	57	16
Findings	784	309
Conclusions	251	91

A statistical model of each type of report is created during system development. The model includes the statistics of: 1) the order of sections within a report class; 2) the number of characters seen within a particular type of section; 3) the types of communications expressed within these sections (e.g., the "Conclusion" section would communicate the types of medical findings, and their possible etiologies). Table 1 is an example of section size statistics collected from 549 chest radiology reports. If a detected section size is larger than its mean by more than three standard deviations, it is considered that another section coexists.

The section suspected of under-segmentation is then scanned for section-specific vocabulary (i.e., cue words) and clues for section boundary such as paragraph breaks and extra white spaces between sentences. For example, cue words such as "seen", "identified", and "appear" strongly indicate that a "Findings" section is present.

RESULTS

The algorithm was implemented in Java and tested on three corpora as summarized in Table 2. For each corpus truth was determined by manually segmenting the reports for both labeled and unlabeled sections. The number of manually identified section categories for the three corpora were 25, 91, and 16 in the order listed. A large number of section categories represented in Corpus 2 is due to many subsections under "Physical Examination" such as "HEENT", "Cardiovascular", "Respiratory", "Abdomen", "Renal", and "Extremities".

Table 2. Parameters of test corpora from two different institutions.

Corpus	1	2	3
Institution (state)	A (CA)	A (CA)	B (FL)
Department	Radiology (Chest)	Urology	Radiology (CT/MR)
Total number of reports	1,957	2,017	3,563
Total number of sections	17,357	62,392	49,554
No. of section categories	25	91	16
No. of dictating physicians	12	14	8
No. of transcribers	9	6	4
No. of reports in training set	549	1,045	248

Rules, lexical patterns, and statistics were compiled from a training set selected for each corpus. The training sets were not part of the corpora used for algorithm testing. The trained classifiers were then applied to the test corpora. The results of section segmentation were compared to the truth, and each section detected by the algorithm was determined either as correctly identified or incorrectly identified. In addition, penalty was scored if a section in the truth data remained undetected by the algorithm.

A section may be detected and its beginning correctly located. However, if a hidden section existed within this section and yet not discovered, then two errors would occur. First, the ending boundary of the detected section will be overestimated. Secondly, the hidden section will remain undetected. Therefore, in such cases both "incorrectly identified" and "undetected" were scored.

Table 3. Algorithm performance.

Corpus		Number of Sections	Correctly Identified	Incorrectly Identified	Undetected
1	Labeled	11,160	99.1%	0.9%	0.1%
	Unlabeled	6,197	98.3%	0.4%	1.4%
	Overall	17,357	98.8%	0.7%	0.6%
2	Labeled	51,116	97.4%	2.2%	0.4%
	Unlabeled	11,276	96.5%	0.3%	3.4%
	Overall	62,392	97.2%	1.9%	0.9%
3	Labeled	45,267	99.4%	0.2%	0.5%
	Unlabeled	4,287	99.0%	0.3%	0.8%
	Overall	49,554	99.4%	0.2%	0.5%

The results of algorithm evaluation are tabulated in Table 3. The detection rates for labeled sections ranged from 97.4 to 99.4%. For unlabeled sections, the detection rates varied from 96.5 to 99.0%. The overall accuracy rate ranged from 97.2 to 99.4%. For all three corpora the detection rates were slightly higher for the labeled sections compared with the unlabeled sections.

DISCUSSION

As shown in Table 3 the algorithm performance varied somewhat according to the type of reports analyzed. The best performance was achieved for Corpus 3 (CT/MR reports), which had the smallest number of section categories. The report format of this corpus was relatively simple and consisted mainly of labeled sections. Hidden sections were present but were highly predictable and readily detectable with the lexical pattern recognition. Because of its structural regularity only a small training set (248 reports) was needed.

Corpus 1 (Chest radiology) contained a fair number of unlabeled sections that were hidden in other sections. Most commonly, the "Findings" sections were subsumed within the "History" sections. This is most likely due to inadvertent or habitual omission of the “Findings” section labels. The advantage of our two-pass algorithm was evident in these cases. The first pass algorithm provided a high precision label-based detection. This, in turn, provided high quality context for the second pass, which effectively applied the statistical analysis of section size in uncovering the buried sections.

Corpus 2 (Urology) presented the most challenge as it contained reports completely devoid of section labels. These were typically letters written in narrative format and contained few clinical elements such as “history”, “physical examination”, and “recommendations”. In the absence of paragraph breaks or labels, the algorithm relies solely on the analysis of section-specific vocabulary. However, if these words are scattered outside the sections with which they were originally associated, erroneous segmentation could occur. In order to overcome this problem we are working toward a context sensitive vocabulary analysis.

The rule-based approach worked well for medical documents, which are usually highly structured. The amount of data needed for supervised learning is dependent on the degree of format irregularities among the reports, the need to segment unlabeled sections, and the number of expected section categories. After training on the initial set of data, one can continue to improve the algorithm performance by periodically updating the knowledge base.

Data processing speed on a 1-GHz Pentium-III PC was less than one second per report. As such the proposed algorithm is suited for real-time applications or off-line processing of a large quantity of medical reports.

ACKNOWLEDGEMENT

The authors thank Carol Demise for her effort in tagging the training and test corpora.

REFERENCES

1. Skorochod’ko EF. Adaptive method of automatic abstracting and indexing. In *Proceedings of the IFIP Congress 71*, 1179-1182, 1972.
2. Halliday MAK and Hasan R. Cohesion in English. Longman, London, 1976.
3. Morris J and Hirst G. Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Computational Linguistics* 17:21-48, 1991.
4. Kozima H. Text segmentation based on similarity between words. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, 286-288, 1993.
5. Reynar JC. An automatic method of finding topic boundaries. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, 331-333, 1994.
6. Hearst MA. Multi-paragraph segmentation of expository text. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, 9-16, 1994.
7. Hearst MA. TextTilling: Segmenting text into multi-paragraph subtopic passages. Computational Linguistics, 23(1):33-64, 1997.
8. Passonneau RJ and Litman DJ. Intention-based segmentation: Human reliability and correlation with linguistic cues. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, 148-155, 1993.
9. Beeferman D, Berger A, and Lafferty J. Text segmentation using exponential models. In *Proceedings of the 2nd Conference on Empirical Methods in Natural Language Processing*, 35-46, 1997.
10. Pevzner L and Hearst MA. A critique and improvement of an evaluation metric for text segmentation. *Computational Linguistics*, 28(1):19-36, 2002.
11. Kaszkiel M and Zobel J. Passage retrieval revisited. In *Proceedings of the 20th International Conference on Research and Development in Information Access*, 178-185, 1997.
12. Mittal V, Kantrowitz M, Goldstein J, and Carbonell J. Selecting text spans for document summaries: Heuristics and metrics. In *Proceedings of the 16th Annual Conference on Artificial Intelligence*, 467-473, 1999.
13. Karlgren J. Stylistic variation in an information retrieval experiment. In *Proceedings of the 2nd International Conference on New Methods in Language Processing*, 1996.
14. Manni I, House D, Maybury M, and Green M. Towards content-based browsing of broadcase news video. In Intelligent Multimedia Information Retrieval. AAAI/MIT Press, pp. 241-258, 1997.